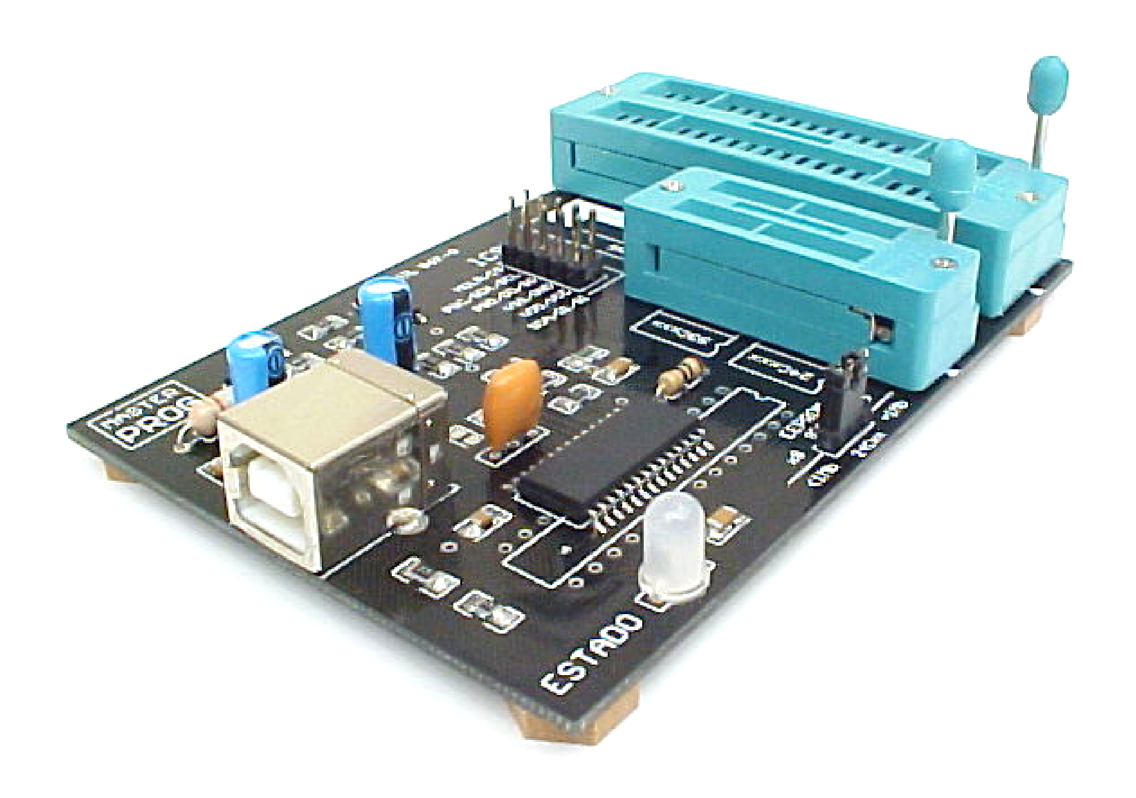
MASTER-PROG USB



Introducción:

El programador MASTER-PROG USB fue diseñado para brindar un óptimo desempeño y versatilidad de uso con el puerto USB (versiones 1.1 y 2.0), su controlador interno emplea tecnología CMOS y por tanto deben tomarse ciertas precauciones al utilizar el MASTER-PROG USB, entre las cuales están las siguientes:

1. Sujetar el programador de preferencia por sus esquinas y cuando no esté en uso, guardarlo en su bolsa antiestática o en su defecto en papel aluminio (¡¡no olvidar quitarlo antes de conectar a la computadora!!).

2. No frotar la placa del programador contra el pelo o la ropa (especialmente si es sintética, como Nylon, Lycra, Poliéster, etc.)

3. Si bien el puerto USB permite la conexión / desconexión de dispositivos sin apagar la computadora, se debe evitar que el programador sea conectado y/o desconectado rápidamente, se deben dejar pasar unos 10 segundos entre desconexiones para dejar que el sistema operativo reconozca al dispositivo y no se produzcan errores o daños al circuito.

4. Evitar conectar el Master-Prog a la computadora cuando tenga un PIC de bajo voltaje (familias PIC18FxxJxx, dsPIC33Fxxx, PIC24Fxxxx, dsPIC30Fxxxx SMPS o EEPROM) insertado en las bases ZIF o conectado al puerto ICSP+, pues puede **dañarse**, lo recomendable es conectarlo al puerto ICSP+ (con alimentación externa de 3.3V) **después** de haber conectado el programador a la PC y de haber seleccionado el tipo de familia de PIC o EEPROM correcta (según corresponda), esto se hace usando el menú <Dispositivo>. Esto ocurre por que el voltaje de programación (VPP/MCLR) por defecto es de 12V (el que usan la mayoría de los PIC's), pero el voltaje para las familias de bajo voltaje es de 4.5V

5. El MASTER-PROG está preparado para poder intercambiar los PIC's sin necesidad de ser desconectado del puerto, sin embargo si el chasis de tu computadora carece de una buena conexión a tierra, podría producirse un daño permanente al PIC y/o al programador, por lo tanto te recomendamos tocar alguna parte metálica del chasis de la computadora (para descargar la electricidad estática) cada vez que cambies de PIC (lo ideal es usar una pulsera antiestática).

Instalación del controlador:

Los pasos a seguir para la instalación del controlador genérico USB, del programador son los siguientes:

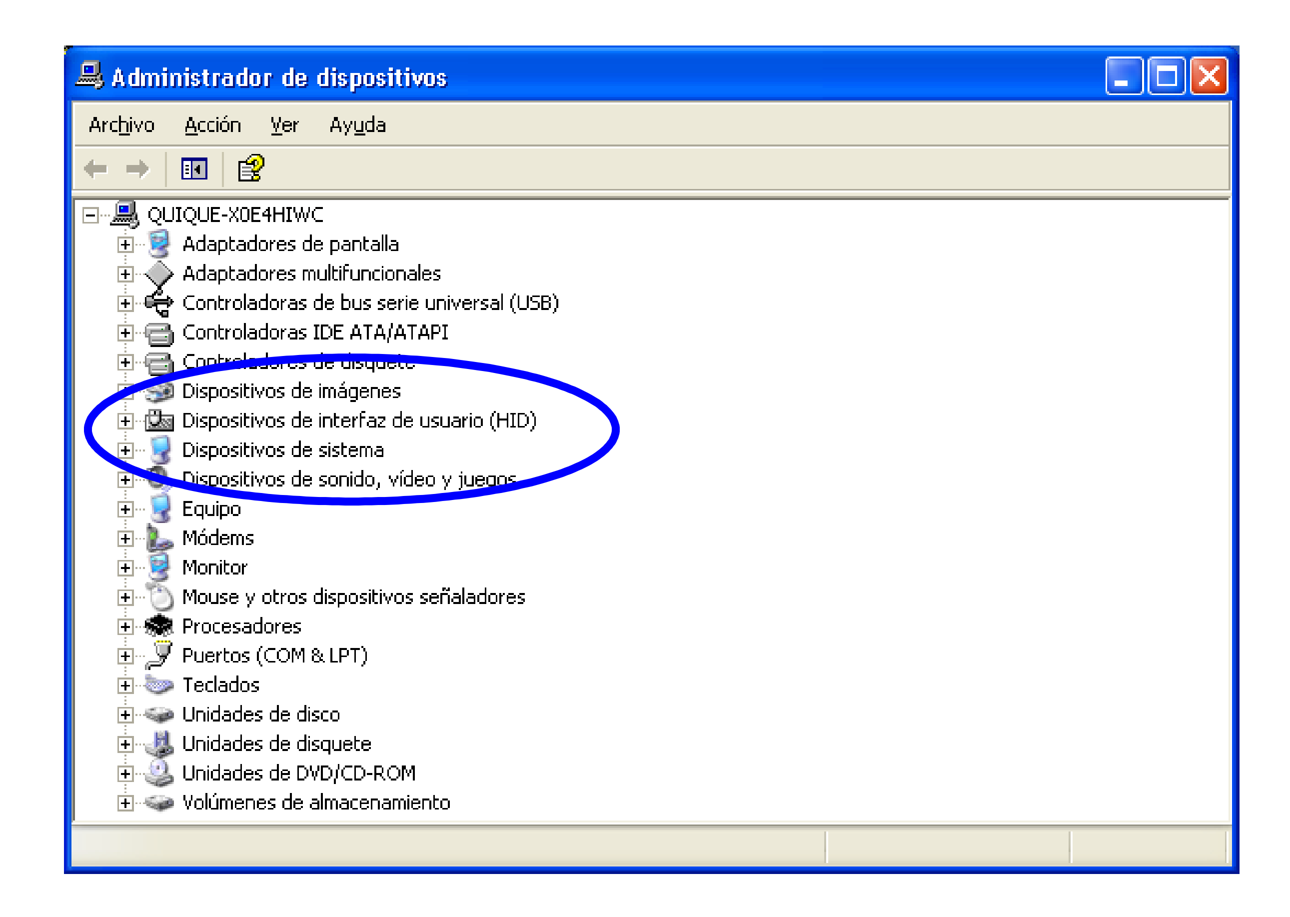
1. *Arrancar la computadora y esperar a la carga completa del sistema operativo.*

2. *Instalar **Framework .Net 2.0**, que está incluido en el CD (en caso de problemas consultar el archivo Leer.txt). Nota:La nueva versión 2.0 del software del MASTER-PROG USB está basada en Framework .Net 2.0, por lo cual podría haber problemas de compatibilidad con los sistemas operativos anteriores a Windows XP, como por ejemplo Windows 98SE, Me, 2000 SP1 por lo cual puede ser necesario descargar e instalar actualizaciones, para dichas versiones de Windows, desde la página de Microsoft www.microsoft.com (Win Vista ya tiene el Framework.Net, consultar las intrcucciones en la carpeta "Inst_Vista")*

3. *Conectar el cable del programador a cualquier puerto USB libre (es compatible con las versiones 2.0 y 1.1), en ese momento encenderá el LED* **"ESTADO"** *de color ROJO (en caso de que esto no suceda, revisar las conexiones, en computadoras ensambladas utilizar únicamente los enchufes USB de la parte posterior de la computadora).*

4. *Esperar a que el sistema operativo reconozca el programador como un dispositivo HID (Human Interface Device).El driver se instalará automáticamente.*

5. *Al finalizar la instalación del driver, debe aparecer la categoría **HID** en el "Administrador de Dispositivos", como se muestra en la siguiente figura (usar:* panel de control > propiedades del sistema*):*



<u>Instalación del Software de Operación:</u>

Para la instalación del software y puesta en marcha del MASTER-PROG USB, sigue estos pasos:

1. Copiar la carpeta "MASTER-PROG" al directorio raíz del disco duro.

2. Quitar el atributo de "Sólo Lectura" a todos los archivos de la carpeta copiada (usar el botón derecho del mouse y la opción propiedades)En el escritorio crear un acceso directo al programa ejecutable "MASTER-PROG.exe".



3. Conecta el programador y espera unos 10 segundos a que el sistema operativo lo reconozca (proceso llamado enumeración).

4. Ejecutar el programa con el acceso directo.

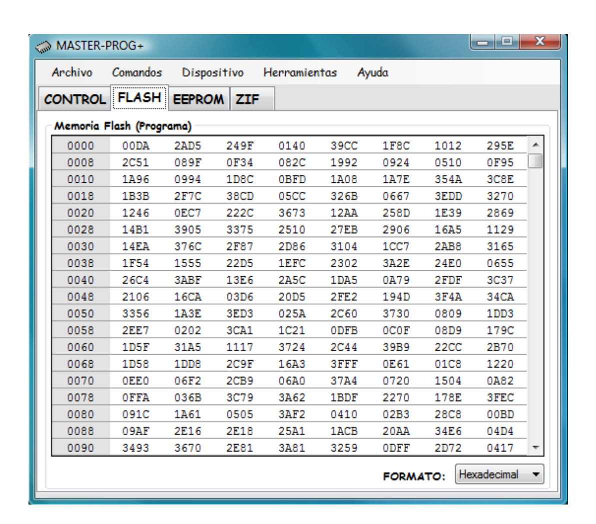
5. Después de parpadear y cambiar de color el indicador "**ESTADO**", debe aparecer la ventana principal (*en este momento **NO** debe haber ningún PIC conectado*):



6. En la ventana de configuración del PIC se muestran los siguientes datos: Dispositivo, ID's del Usuario, OSCCAL y los Bits de Configuración. Debido a que no hay un PIC conectado se muestra el mensaje "No Detectado" (este mensaje también se produce en caso de conectar un PIC que no sea compatible con el programador, para más información consultar la lista de dispositivos soportados).

7. Seleccionar del menú la familia correcta usando <***Dispositivo***>. *<u>IMPORTANTE</u>: leer las precauciones que se mencionan en la introducción,sobre todo con los dispositivos de bajo voltaje* (véase: *<u>lista de dispositivos soportados</u>*).
8. Insertar un PIC en la base ZIF y hacer clic en el botón <**LEER**>, después de unos segundos se muestra en *Dispositivo:* el número del PIC insertado, por ejemplo: PIC16F877A y el contenido de memoria (si está programado). Si se muestra el mensaje *"¡PIC NO soportado!"* revisar si el PIC está en la lista de dispositivos soportados o si insertó de forma correcta. Volver a intentar si es necesario.

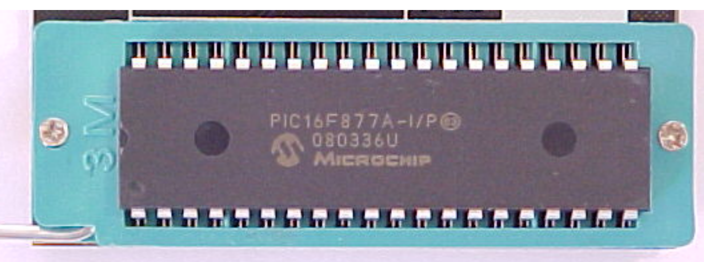




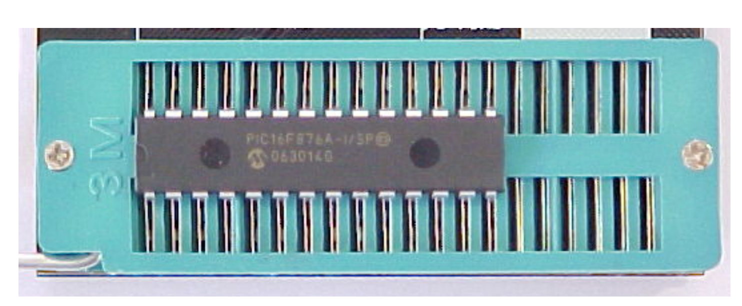
9. Para finalizar usar los botones <**LEER**> o <**VERIFICAR**> para comprobar la comunicación entre la PC y el programador, el indicador LED **"ESTADO"** debe cambiar de color (Rojo/Amarillo) indicando que existe transferencia de datos.

Colocación Correcta de los PIC’s:

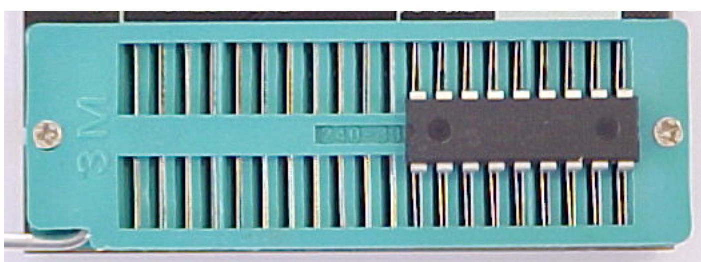
a) 40 pines:



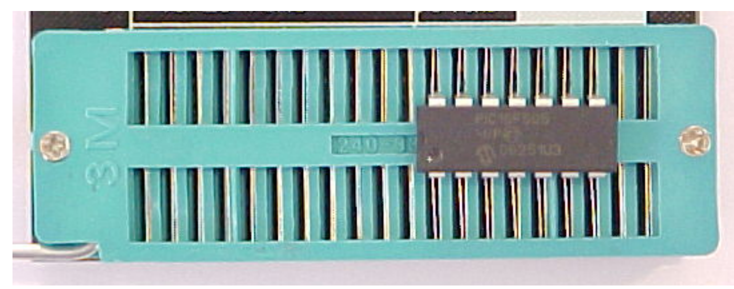
b) 28 pines:



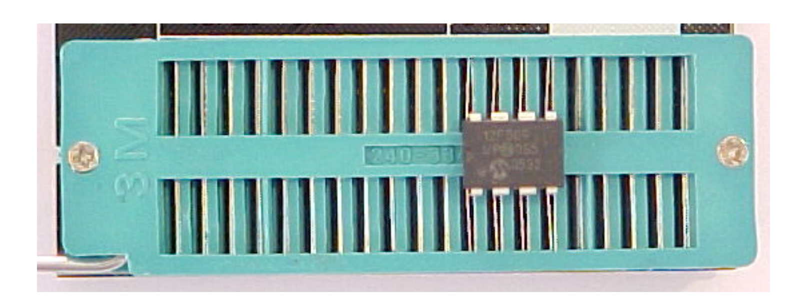
c) 18 pines:



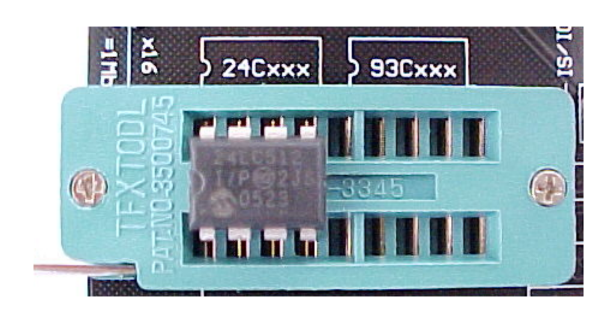
d) 14 pines:



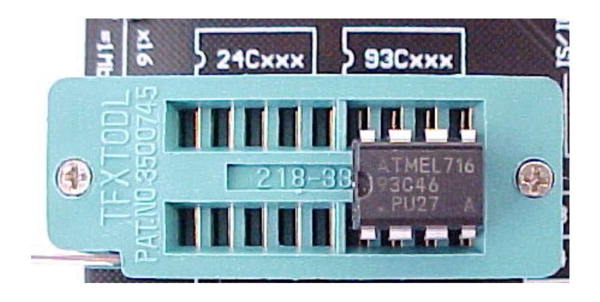
e) 8 pines:



f) EEPROM 24Cxxx:



g) EEPROM 93Cxxx:



***NOTA:* La familia 10Fxxx tiene encapsulado de 8 pines, pero es incompatible con las conexiones de la base (diseñada para la familia 12Fxxx y 16Fxxx), por lo cual se debe utilizar el puerto ICSP (Ver archivo adjunto sobre este puerto). Lo mismo ocurre con los dsPIC's de 40, 28 y 18 pines, que por sus características eléctricas, deben conectarse también con el puerto ICSP+.**

Comandos del Menú:

Los comandos que se encuentran en la barra del menú tienen las siguientes funciones:

a) **<Archivo>.**
Abrir *.hex: Abre le archivo de extensión ".hex" que es cargado en el PIC (utilizado con el comando ***Escribir PIC***), dicho archivo debe incluir los bits de configuración (Watchdog, tipo de oscilador, protección de código, etc.), de lo contrario se recibe el mensaje "Bits de Configuración NO encontrados", en este caso el programa puede NO ser ejecutado correctamente. Para corregir esto se debe usar la directiva "__CONFIG" en el encabezado del programa ensamblador antes de compilarlo, como se muestra en ejemplo:

```
; ************ ENCABEZADO ***********

#INCLUDE        <P16F877A.INC>

                __CONFIG        _WDT_OFF & _XT_OSC & _LVP_OFF & _CP_OFF & _CPD_OFF & _BODEN_OFF

                        LIST    P=16F877A
DLY_1           EQU             20h             ; REGISTRO PARA RETARDO
DLY_2           EQU             21h             ; REGISTRO PARA RETARDO
```

Directivas de __CONFIG:

Feature	SYMBOLS
Oscillators	_RC_OSC
	_EXTRC_OSC
	_EXTRC_OSC_CLKOUT
	_EXTRC_OSC_NOCLKOUT
	_INTRC_OSC
	_INTRC_OSC_CLKOUT
	_INTRC_OSC_NOCLKOUT
	_LP_OSC
	_XT_OSC
	_HS_OSC
Watch Dog Timer	_WDT_ON
	_WDT_OFF
Power-up Timer	_PWRTE_ON
	_PWRTE_OFF
Brown-out Reset	_BODEN_ON
	_BODEN_OFF
Master Clear Enable	_MCLRE_ON
	_MCLRE_OFF
Code Protect	_CP_ALL
	_CP_ON
	_CP_75
	_CP_50
	_CP_OFF
Code Protect Data EEPROM	_DP_ON
	_DP_OFF
Code Protect Calibration Space	_CPC_ON
	_CPC_OFF

Para más detalles consultar los manuales del MPLAB IDE, Lenguaje MPASM y las fichas técnicas del PIC utilizado.
Otra forma de incluir los Bits de Configuración, es ajustarlos manualmente por medio del menú **Configure** > **Configuration Bits**, como se muestra en la figura:

Temp_F877 - MPLAB IDE v7.41

File Edit View Project Debugger Programmer Tools Configure Window Help

Checksum: 0x07ff

Temp_F877.mcp
Source Files
Temp_F877.asm
Header Files
Object Files
Library Files
Linker Script
Other Files

Files Symbols

C:\PROYECTOS\TEMP_F877\TEMP_F877.ASM

```
; PROGRAMA PARA UTILIZAR EL CONVERTIDOR
; ANALÓGICO-DIGITAL (ENTRADA=AN0), PARA MEDIR LA TEMPERATURA
; Y MOSTRARLA EN LA PANTALLA LCD.
; ****************** CONEXIÓN DEL LCD ******************
;  DB7..DB4=RB7..RB4,  DB3..DB0=GND,  RS=RB1,  E=RB0
TEMP        EQU        23h        ; REGISTRO T
CEN              EQU        24h        ; CE
DEC              EQU        25h        ; DE
UNI              EQU        26h        ; UN

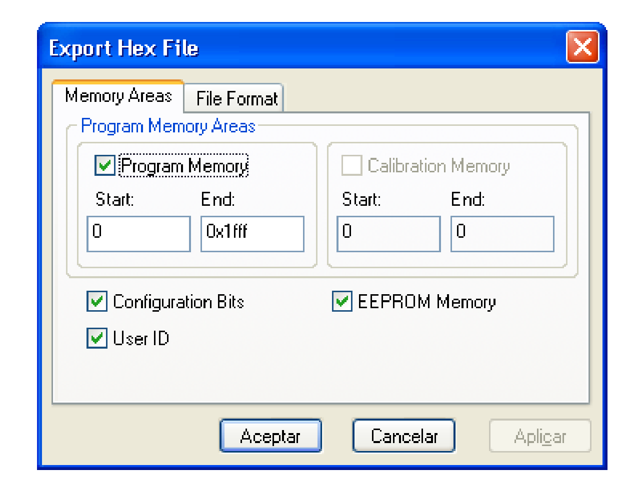
; ************* CONFIGURAR PUERTOS ***************
```

Configuration Bits

Address	Value	Category	Setting
2007	0800	Oscillator	LP
		Watchdog Timer	Off
		Power Up Timer	On
		Brown Out Detect	Off
		Low Voltage Program	Disabled
		Flash Program Write	0h to FFFh write Protected
		Data EE Read Protect	On
		Code Protect	On

MPLAB SIM | MPLAB ICD 2 | PIC16F877A | pc:0 | W:0 | z dc c | bank 0

Después utilizando la opción del menú **File** > **Export...** en MPLAB, se guarda el archivo compilado *.hex.



Guardar *.hex: Guarda el contenido de la memoria del PIC (Flash, EEPROM y Configuración) obtenido por medio del comando <**LEER**>, esto permite tener un respaldo del contenido interno del PIC y utilizarlo para crear varias copias de un mismo programa. En caso de que el PIC este protegido contra lectura sólo se leerán “0000” de la memoria (haciendo imposible su copia).

b) **<Dispositivo>.**

Familia ***Baseline*** (Series 10Fxxx, 12F5xx, 16F50x): Selecciona la familia de PIC’s con núcleo de 12 bits, entre los cuales están el 12F508, 16F505, 10F200, etc. (éstos deben seleccionarse de forma manual).

Familia ***Midrange*** (Serie 16Fxxx, 12F5xx, 16F50x): Selecciona la familia de PIC’s con núcleo de 14 bits, entre los cuales están el 16F628A, 16F873A, 16F877A, etc. (El número de PIC es seleccionado automáticamente al realizar cualquier operación, por ejemplo lectura, checar programador, etc.)

Familia ***PIC18F y PIC18F_K_***: Selecciona la familia de PIC’s de alto desempeño (llamados también *High performance*), entre los cuales están el 18F452A, 18F4539A, etc. (El número de PIC es seleccionado automáticamente al realizar cualquier operación, por ejemplo lectura, verificación, etc.)

Familia ***PIC18F_J_***: Selecciona la familia de PIC’s de alto desempeño y bajo voltaje (3.3V), entre los cuales están el 18F24J10, 18F65J10, etc. (Nota: Debido a que estos PIC se alimentan con 3.3V, requieren alimentación externa y deben conectarse al programador por medio del puerto ICSP, pero sin usar VDD.)

Familia ***PIC24***: Selecciona la familia de PIC’s con arquitectura interna de 16 bits, entre los cuales están el PIC24FJ64GA008, 18F65J10, PIC24FJ128GA010, etc. (Nota: Debido a que estos PIC se alimentan con 3.3V, requieren alimentación externa y deben conectarse al programador por medio del puerto ICSP,

pero sin usar VDD.)

Familia ***dsPIC30*** : Selecciona la familia de dsPIC's con arquitectura interna de 16 bits (Nota: Estos dsPIC's deben conectarse al programador por medio del puerto ICSP, usando las 5 terminales)

Familia ***dsPIC33 y dsPIC30 SMPS*** : Selecciona la familia de dsPIC's con arquitectura interna de 16 bits (Nota: Debido a que estos dsPIC's se alimentan con 3.3V, requieren alimentación externa y deben conectarse al programador por medio del puerto ICSP+, pero sin usar VDD.)

Familia ***KEELOQ HCS*** : Selecciona la familia de dispositivos de seguridad electrónicos "KEELOQ", los cuales funcionan como "cerraduras electrónicas" en aplicaciones como alarmas, sistemas de seguridad, control de acceso, etc. Se deben programar por medio del puerto ICSP+

Familia ***EEPROMS*** : Selecciona la familia de memorias EEPROM seriales de tipo I^2C y Microwire de las series 24xxx, 25xxx, 93xxxA (8 bits), 93xxxB (16 bits) y 93xxxC (8/16 bits). Las familias 24xxx y 93xxx son soportadas por la base ZIF de 18 pines y el puerto ICSP+. La familia 25xxx se debe programar por medio del puerto ICSP+. El "JUMPER" del programador permite seleccionar el formato de la memoria 93xxxC entre 8 ó 16 bits. Si la memoria es de tipo I^2C (serie 24xxx) el "JUMPER" se usa para seleccionar el tamaño en bits, entre 1Mb como la 24C1025 y menor de 1Mb (512, 256, 128, etc.)

c) **<Comandos>.**

LEER: Lee el contenido total de la memoria del PIC (Flash, EEPROM y Configuración) y lo muestra en el área del buffer de la ventana principal, éste puede ser guardado o bien reescrito en otro PIC nuevo (simplemente se debe retirar el PIC fuente de la base del programador y colocar el PIC destino.)

ESCRIBIR: Escribe el contenido del buffer en la memoria del PIC, el cual puedo ser obtenido de un archivo *.hex o bien de la

lectura de otro PIC.

VERIFICAR: Verifica (compara) el contenido del buffer con el contenido de la memoria del PIC, en caso de existir alguna diferencia se produce un mensaje de error, esto se puede producir por un daño del PIC o bien por que el programa tiene la opción de protección de código (protección contra lectura.)

BORRAR: Borra el contenido total de la memoria del PIC, sin embargo, en la familia Baseline se salvan los valores del OSSCAL y el BandGap (consultar sus fichas técnicas para mayor información.)

¿DISPOSITIVO BORRADO?: Comprueba si la memoria del PIC está en blanco (Borrada), si encuentra algún dato en la memoria (Flash/EEPROM) devuelve la dirección donde se localiza en la barra de estado.

Verificar al Escribir: Al activarse esta opción verifica los datos que se están escribiendo en el PIC al momento de programarlo, en caso de existir un error se detendrá el proceso de escritura y se muestra un mensaje de error en la barra de estado, esto reduce la velocidad de escritura, pero asegura que los datos escritos en el PIC sean correctos.

d) **<Herramientas>**.
Proteger Código (Anti Copia): Habilita la función del Bit de protección contra lectura del PIC (en caso de que el programa fuente no la tenga activada), al escribir el PIC con esta opción, se evita la lectura y copia del contenido de la memoria Flash (se leen “0000”), los datos de la memoria EEPROM si se pueden leer. Esta opción produce un error de verificación.

Proteger Datos (Memoria EEPROM): Permite habilitar la función del Bit de protección contra lectura del PIC (en caso de que el programa fuente no la tenga activada), al escribir el PIC con esta opción, se evita la lectura y copia del contenido de la memoria EEPROM (se leen “00”), los datos de la memoria Flash si se pueden leer. Esta opción produce un error de verificación de la memoria EEPROM.

Ajustar OSSCAL (Oscilador Interno): Ajusta manualmente el valor del OSCCAL en los dispositivos que cuentan con oscilador interno RC, haciendo que su frecuencia cambie (para más detalles ver la ficha técnica del PIC utilizado).

Programación Rápida: Acelera las funciones de Escritura/Lectura del PIC, se recomienda dejarla activada. En caso de que aparecieran errores puede ser necesario desactivarla, especialmente si se usa el puerto ICSP.

AUTO/CONEX: Tiene la función de revisar el funcionamiento interno del MASTER-PROG USB y la comunicación con la PC, en caso de alguna anomalía se produce un mensaje de error (si esta condición persiste comunicarse con nosotros para recibir asesoría). *IMPORTANTE: En ciertos tipos de PIC's es posible que se produzca un* ***DAÑO*** *al usar esta función (esto ocurre en dispositivos de bajo voltaje), por lo tanto se recomienda retirar cualquier PIC insertado en las bases del Master-Prog. Leer las advertencias mostradas en la introducción. Nota: Esta función puede borrar o alterar el contenido del PIC, no usar con PIC's preprogramados, primero seleccionar la familia correcta y luego insertar el PIC en la base ZIF o conectarlo al ICSP para poder usar las funciones Leer, Verificar, etc.*

e) **<Ayuda>**.
Muestra las diferentes opciones de ayuda en línea y el soporte técnico.

Modo *AUTOPROG*

Esta función especial del MASTER-PROG te permite enlazar cualquier software de compilación, que genere los archivos *.hex por ejemplo el MPLAB (en ensamblador o "C"), para programar automáticamente el PIC cada vez que se compila el código fuente. El ***AUTOPROG*** monitorea los cambios o actualizaciones del archivo *.hex al momento de compilar y si el PIC está conectado al programador, el nuevo *.hex es descargado y verificado en el PIC aun minimizado el software del MASTER-PROG.

Para usarlo sigue estos pasos:

■ Corre tu software de compilación, compila tu programa fuente y genera por primera vez el archivo *.hex

■ Conecta el MASTER-PROG y corre el software de programación, selecciona la familia correcta de PIC o dsPIC y coloca el dispositivo a programar en la base (o conéctalo por medio del puerto ICSP+).

■ Haz clic en el botón “Activar/Salir AUTOPROG”.

■ Selecciona el archivo *.hex que vas a autoprogramar en el PIC (el mismo que generó el compilador).

■ El *.hex será programado en el PIC y el software queda en modo de “espera”.

■ Cada vez que recompilas tu código el software programará automáticamente el PIC (el LED estado cambiará de color), retira el PIC de la base cuando el LED esté en **ROJO** (si usas el puerto ICSP+ no es necesario desconectarlo).

■ Para detener el modo ***AUTOPROG*** haz clic en el botón “Activar/ Salir AUTOPROG”.

Para Mayor Información Escribe a:

edutronika@hotmail.com